

Assignment 6-6 Polynomial Addition 4

You are given two polynomials $f(x)$ and $g(x)$ with integer coefficients. In this problem you'll have to find out the sum $f(x) + g(x)$ of $f(x)$ and $g(x)$. For the sake of convenience, we define the zero polynomial as $0x^0$.

Input

The input consists of t ($30 \leq t \leq 40$) test cases. The first line of the input contains only positive integer t . Then t test cases follow. Each test case consists of two polynomials $f(x)$ and $g(x)$. Each polynomial consists of three lines: the first line contains only one positive integer m ($m \leq 100$) which represents the number of nonzero terms of the polynomial; the second line consists of m nonzero integers (in the range $[-2^{62}, 2^{62} - 1]$) representing the coefficients of the polynomial; the third line consists of m integers (in the range $[0, 2^{62} - 1]$ and in decreasing order) representing the corresponding exponents of the polynomial. For example, the polynomial $300x^{1000} + 200x^{100} + 100x^{10}$ is represented as the following three lines:

```
3
300 200 100
1000 100 10
```

You may assume that neither $f(x)$ nor $g(x)$ is the zero polynomial.

Output

For each test case, you are to output three lines representing $f(x) + g(x)$, in the same format as the input. Note that if $f(x) + g(x)$ is the zero polynomial, the output should be as follows:

```
1
0
0
```

Sample Input

```
2
3
1 -2 -4
3 2 0
1
4
0
3
```

```

-1 -2 4
3 2 0
4
1 2 1 -7
3 2 1 0

```

Sample Output

```

2
1 -2
3 2
2
1 -3
1 0

```

Requirements

In your program, a polynomial $a_n x^{d_n} + a_{n-1} x^{d_{n-1}} + \dots + a_2 x^{d_2} + a_1 x^{d_1} + a_0 x^{d_0}$ should be represented by the following two arrays:

$$\begin{array}{cccccc}
 & n & n-1 & \cdots & 2 & 1 & 0 \\
 \text{coefficient array: } & \boxed{a_n} & \boxed{a_{n-1}} & \boxed{\cdots} & \boxed{a_2} & \boxed{a_1} & \boxed{a_0} \\
 \\
 & n & n-1 & \cdots & 2 & 1 & 0 \\
 \text{exponent array: } & \boxed{d_n} & \boxed{d_{n-1}} & \boxed{\cdots} & \boxed{d_2} & \boxed{d_1} & \boxed{d_0}
 \end{array}$$

For example, the polynomial $-x^3 - 2x^2 + 4$ is represented by the following two arrays:

$$\begin{array}{cccc}
 & 2 & 1 & 0 \\
 \text{coefficient array: } & \boxed{-1} & \boxed{-2} & \boxed{4} \\
 \\
 & 2 & 1 & 0 \\
 \text{exponent array: } & \boxed{3} & \boxed{2} & \boxed{0}
 \end{array}$$

Note that $d_n > d_{n-1} > \dots > d_2 > d_1 > d_0 \geq 0$, and for every $i = 0, 1, \dots, n$, $a_i \neq 0$.

Part of the program

You are required to write the function `addition` to complete the following program which solves this problem. In your program, you cannot declare global variables or static arrays.

```

// Polynomial addition
#include <iostream>
using namespace std;

// sum = addend + adder
void addition( long long int *addendCoef, long long int *addendExpon, int
addendSize,
               long long int *adderCoef, long long int *adderExpon, int
adderSize,
               long long int *&sumCoef, long long int *&sumExpon, int &sumSize );

// outputs the specified polynomial
void output( long long int *coefficient, long long int *exponent, int size );

int main()
{
    int T;
    cin >> T;
    for( int t = 0; t < T; t++ )
    {
        int addendSize;
        cin >> addendSize;
        long long int *addendCoef = new long long int[ addendSize ];
        long long int *addendExpon = new long long int[ addendSize ];
        for( int i = addendSize - 1; i >= 0; i-- )
            cin >> addendCoef[ i ];
        for( int i = addendSize - 1; i >= 0; i-- )
            cin >> addendExpon[ i ];

        int adderSize;
        cin >> adderSize;
        long long int *adderCoef = new long long int[ adderSize ];
        long long int *adderExpon = new long long int[ adderSize ];
        for( int i = adderSize - 1; i >= 0; i-- )
            cin >> adderCoef[ i ];
        for( int i = adderSize - 1; i >= 0; i-- )
            cin >> adderExpon[ i ];

        int sumSize;
        long long int *sumCoef;
        long long int *sumExpon;
        addition( addendCoef, addendExpon, addendSize,
                  adderCoef, adderExpon, adderSize,
                  sumCoef, sumExpon, sumSize );

        output( sumCoef, sumExpon, sumSize );

        delete[] addendCoef;
        delete[] addendExpon;
        delete[] adderCoef;
        delete[] adderExpon;
        delete[] sumCoef;
        delete[] sumExpon;
    }
}

// sum = addend + adder
void addition( long long int *addendCoef, long long int *addendExpon, int
addendSize,
               long long int *adderCoef, long long int *adderExpon, int
adderSize,
               long long int *&sumCoef, long long int *&sumExpon, int &sumSize )
{

}

// outputs the specified polynomial
void output( long long int *coefficient, long long int *exponent, int size )
{
    cout << size << endl;
}

```

```
cout << coefficient[ size - 1 ];  
for( int i = size - 2; i >= 0; i-- )  
    cout << " " << coefficient[ i ];  
cout << endl;  
  
cout << exponent[ size - 1 ];  
for( int i = size - 2; i >= 0; i-- )  
    cout << " " << exponent[ i ];  
cout << endl;  
}
```